Claims

[Claim 1] An apparatus comprising:

a) a plurality of memory elements, each of which comprising:
   1) at least one register, comprising at least one addressable register;
   2) an instruction input;
   3) executing means for carrying out instructions for the memory element from its instruction input;
   4) an enable bit input;
   5) enabling means for enabling its executing means only when its enable bit input is positively asserted; and
   6) a unique element address;

b) an input/output control unit, comprising:
   1) an external bus connection, and
   2) means for communicating with the external bus;

c) exclusive means for exclusively copying either (1) the content of any addressable register to its exclusive bus, or (2) the content of its exclusive bus to any addressable register;

d) concurrent means for concurrently executing a same instruction in all its enabled memory elements, the concurrent means further comprising:
   1) enabling means for concurrently positively asserting the enable bit inputs of all memory elements whose element addresses are (A) no less than a start element address, (B) no more than a end element address, and (C) an integer increment of a carry number starting from the starting element address; and
   2) instructing means for sending a instruction concurrently to all memory elements; and

e) instruction means for receiving and carrying out instructions at the external bus connections of the apparatus, further comprising:
   1) command means for translating the content at the external bus connections of the apparatus into exclusive means and concurrent means;
   2) result means for presenting the result of the command means at the external bus connections of the apparatus.

[Claim 2] An apparatus of [Claim 1], while performing a concurrent means, further comprising parallel processing means for simultaneously performing an exclusive means on any addressable registers which is not involved in the on-going concurrent means.

[Claim 3] An preferred implementation of apparatus of [Claim 2], further comprising:

a) every register in each of all its memory elements is addressable; and

b) steps for using the parallel processing means to update the addressable registers for other tasks which are different with the task that involves the on-going concurrent means, so that future task switching is instant.

[Claim 4] An apparatus of [Claim 1], its instruction means further comprising termination means for signaling the termination of the instruction means by:

a) means for changing the content of the external bus of the apparatus in a predefined way, or
b) means for waiting a predefined time period before able to receive another input from the external bus connections, or
c) the combination of (a) and (b).

[Claim 5] An apparatus of [Claim 1],
a) its input/output control unit further comprising a command bit input; and
b) its instruction means further comprising memory means for behaving as a conventional random access memory when the command bit input is negatively asserted.

[Claim 6] An apparatus of [Claim 5], its input/output control unit further comprising compliance means for making the connection to the external bus of the in full compliance with a bus standard, the compliance means further comprising:
a) means for connecting with an external bus which complies with the bus standard comprising (1) address bus, (2) data bus and (3) control bus;
b) means for making the apparatus' external bus connections to the data bus in full compliance with the data bus portion of the bus standard;
c) means for making the apparatus' external bus connections to the address bus in full compliance with the corresponding bits of the address bus portion of the bus standard;
d) means for making the apparatus' command bit input in full compliance with a bit of the address bus of the bus standard which is not used to connect to the apparatus' connections to the address bus, as if the address bus bit of the bus standard is being used as a address bus bit; and
e) means for making the apparatus' external bus connections to the control bus in full compliance with the bits or bits' logic combinations of the control bus portion and the remained unconnected bits of the address bus portion of the bus standard.

[Claim 7] Preferred compliance means for the apparatus' connection to the external bus in full compliance with a bus standard as claimed in [Claim 6] the preferred compliance means further comprising connecting the apparatus' command bit input with the least significant address bit of the bus standard which is not connected to the apparatus' external bus connection to the address bus.

[Claim 8] Using steps for using the apparatus when it is connected with other devices using an external bus of a bus standard, as claimed in [Claim 6], the using steps comprising:
a) negatively asserting the command bit input of the apparatus, to use the apparatus as a conventional random access memory,
b) positively asserting the command bit input of the apparatus, and sending a processing instruction to the apparatus as if storing data to a fictional location inside the apparatus, and
c) positively asserting the command bit input of the external bus, and sending a characterizing instruction to the apparatus as if retrieving data from a fictional location inside the apparatus.

[Claim 9] An apparatus of [Claim 1], its instruction means further comprising:

a) an instruction kernel, storing predefined combinations of instructions;
b) means for translating the content of its external bus into instructions for the apparatus using its instruction kernel;
c) means for signaling the content of the output portion of its external bus being either valid or invalid for the current content of its external bus;
d) means for carrying out the instruction for the apparatus in a series of steps comprising the concurrent means and the exclusive means which are stored in the instruction kernel; and
e) means for using an existing bus standard protocol to signal the readiness of the apparatus.

[Claim 10] An apparatus of [Claim 1], the value of the carry number of the enabling means of its concurrent means being no larger than the square root of its total memory element count.

[Claim 11] An apparatus of [Claim 1], further comprising:

a) a general decoder, comprising:
    1) a start address input;
    2) an end address input;
    3) a carry number input;
    4) a plurality of bit outputs, each of which has a unique address; and
    5) means for concurrently positively asserting all the bit outputs whose addresses are: (A) no less than the value at the start address input, (B) no more than the value at the end address input, and (C) an integer increment of the value at the carry number input starting from the value at the start address input, while negatively asserting all the other bit outputs; and
b) the enabling means of its concurrent means further comprising means for using the general decoder to implement the enabling of all memory elements whose element addresses are (1) no less than a start element address, (2) no more than an end element address, and (3) an integer increment of a carry number starting from the starting element address.

[Claim 12] An apparatus of [Claim 1], the value of the carry number of the enabling means of its concurrent means being a constant one, so that all memory elements within an address range are enabled.

[Claim 13] An apparatus of [Claim 12], further comprising:

a) a range decoder, comprising:
    1) a start address input;
    2) an end address input;
    3) a plurality of bit outputs, each of which has a unique address; and
    4) means for concurrently positively asserting all the bit outputs whose addresses are: (A) no less than the value at the start address input, and (B) no more than the

value at the end address input, while negatively asserting all the other bit outputs; and

b) the enabling means of its concurrent means further comprising means for using the range decoder to implement the enabling of all memory elements within an address range between (1) a start address and (2) an end address.

[Claim 14] An apparatus comprising:

a) a plurality of memory elements, each of which comprising:
   1) at least one register, comprising:
      A) at least one addressable register; and
      B) a neighboring register;
   2) an instruction input, comprising a select bit;
   3) executing means for carrying out instructions for the memory element from its instruction input;
   4) an enable bit input;
   5) enabling means for enabling its executing means only when its enable bit input is positively asserted;
   6) a unique element address;
   7) neighboring means for reading the neighboring register of the neighboring memory element whose element address is either (A) immediately lower when the select bit of its instruction input is negatively asserted, or (B) immediately higher when the select bit of its instruction input is positively asserted.

b) an input/output control unit, comprising:
   1) an external bus connection, and
   2) means for communicating with the external bus;

c) exclusive means for exclusively copying either (1) the content of any addressable register to its exclusive bus, or (2) the content of its exclusive bus to any addressable register;

d) concurrent means for concurrently executing a same instruction in all its enabled memory elements, the concurrent means further comprising:
   1) enabling means for concurrently positively asserting the enable bit inputs of all memory elements whose element addresses are (A) no less than a start element address, (B) no more than a end element address, and (C) an integer increment of a carry number starting from the starting element address; and
   2) instructing means for sending a instruction concurrently to all memory elements; and

e) instruction means for receiving and carrying out instructions at the external bus connections of the apparatus, further comprising:
   1) command means for translating the content at the external bus connections of the apparatus into exclusive means and concurrent means;
   2) result means for presenting the result of the command means at the external bus connections of the apparatus.

[Claim 15] An apparatus of [Claim 14]:

a) each of its memory elements further comprising:

1) one addressable register;
2) one neighboring register;
3) its instruction input further comprising a self bit,
4) self means for copying the content of its addressable register to its neighboring register when the self bit of its instruction input is positively asserted; and
5) neighboring means for copying the content of the neighboring register of its neighboring memory element to its addressable register when the self bit of its instruction input is negatively asserted;

b) its concurrent means further comprising:
1) the carry number of the enabling means is a constant one, so that all memory elements within the address range defined by the starting address and the end address are enabled;
2) up moving means for concurrently moving the content of each of all addressable registers within an address range to the addressable register whose register address is immediately higher; and
3) down moving means for concurrently moving the content of each of all addressable registers within an address range to the addressable register whose register address is immediately lower.

[Claim 16] Steps for using apparatus of [Claim 15], further comprising:
1) allocating steps for concurrently allocating a new memory allocation while keeping all the memory allocations closely packed using the up moving means; and
2) de-allocating steps for de-allocating an existing memory allocation while keeping all the memory allocations closely packed using the down moving means; and
3) changing steps for changing an existing memory allocation while keeping all the memory allocations closely packed using either the up moving means or the down moving means;

[Claim 17] Steps for using apparatus of [Claim 15], further comprising:
1) allocating steps for concurrently allocating a new memory allocation while keeping all the memory allocations closely packed using the down moving means;
2) de-allocating steps for de-allocating an existing memory allocation while keeping all the memory allocations closely packed using the up moving means; and
3) changing steps for changing an existing memory allocation while keeping all the memory allocations closely packed using either the up moving means or the down moving means;

[Claim 18] A preferred implementation of apparatus of [Claim 15], its neighboring registers are made of dynamic memory cells each of which only need to remember its datum for one instruction cycle.

[Claim 19] A preferred implementation of apparatus of [Claim 15]:
a) both its neighboring registers and its addressable registers are made of dynamic memory cells; and

b) its concurrent means further comprising refreshing means for concurrently refreshing the contents of each of all addressable registers by a consecutive allocation and de-allocation of one addressable register.

[Claim 20] An apparatus of [Claim 19] further comprising:
a) a timer; and
b) the refreshing means of its concurrent means further comprising means for using the timer to invoke the refreshing means automatically.

[Claim 21] Means for using apparatus of [Claim 19] at where a static random access memory is traditionally used.

[Claim 22] An apparatus of [Claim 15], further comprising:
a) a data object table comprising entries of data objects, with each entry comprising:
   1) an unique ID; and
   2) memory allocation for the memory object;
b) its concurrent means further comprising:
   1) means for identifying each data objects in the apparatus by a unique object ID which is independent of the memory allocation for the data object in the apparatus;
   2) means for adding a new data object and obtaining the corresponding new object ID;
   3) means for removing a such identified data object;
   4) means for changing the size of a such identified data object;
   5) means for accessing any part of a such identified data object by an offset into the data object; and
   6) means for refusing any of the above requests when the apparatus is unable to properly satisfy a such request.

[Claim 23] An apparatus of [Claim 22], further comprising:
a) each entry of the data object table further comprising
   1) an optional ID of container data object; and
   2) an optional list of IDs of contained data object;
b) its concurrent means further comprising:
   1) means for defining any contained data object within a container data object;
   2) means for adjusting the memory allocation of the container data object accordingly when adjusting the memory allocation of any contained data object of the container data object;
   3) means for removing any defined contained data object within a container data object and adjusting the memory allocation of the container data object accordingly; and
   4) when de-allocating a container data object, means for de-allocating all the contained data object(s) of the container data object.

[Claim 24] An apparatus of [Claim 15], further comprising:
a) each of its memory elements further comprising:
   1) the addressable register further comprising an additional even-or-odd bit;

2) the neighboring register further comprising an additional even-or-odd bit;
3) its self means further comprising means for copying the value of the even-or-odd bit of the addressable register to the even-or-odd bit of the neighboring register; and
4) its neighboring means further comprising means for copying the value of the even-or-odd bit of the neighboring register of the neighboring memory element to the even-or-odd bit of its addressable register; and

b) its exclusive means further comprising:
1) even-or-odd means for calculating the even-or-odd value of the content of any addressable register;
2) when copying the content of its exclusive bus to any of its addressable register, means for assigning the even-or-odd value of the addressable register to the even-or-odd bit of the addressable register; and
3) when copying the content of any of its addressable register to its exclusive bus, means for
A) detecting the disagreement of the even-or-odd bits between the addressable register and its corresponding neighboring register,
B) when the two even-or-odd bits agrees, copying the content of the addressable register to its exclusive bus; and
C) when the two even-or-odd bits disagrees, calculating the even-or-odd value of the content the addressable register, and either (i) copying the content of the addressable register to its exclusive bus if the even-or-odd bit of the addressable register is asserted correctly; or (ii) copying the content of the corresponding neighboring register of the addressable register to its exclusive bus if the even-or-odd bit of the addressable register is not asserted correctly.

[Claim 25] An apparatus comprising:

a) a plurality of memory elements, each of which comprising:
1) at least one register, comprising at least one addressable register;
2) an instruction input;
3) executing means for carrying out instructions for the memory element from its instruction input;
4) an enable bit input;
5) enabling means for only enabling the executing means when the enable bit input is positively asserted;
6) a match bit output;
7) state means for defining states for the memory element; and
8) matching means for positively asserting its match bit output only when the memory element is in a required state;

b) an input/output control unit, comprising:
1) a external bus connection, and
2) means for communicating with the external bus;

c) exclusive means for exclusively copying either (1) the content of any addressable register to its exclusive bus, or (2) the content of its exclusive bus to any addressable register;

d) concurrent means for concurrently executing a same instruction in all its enabled memory elements, the concurrent means further comprising:
   1) enabling means for concurrently positively asserting the enable bit inputs of all memory elements whose element addresses are (A) no less than a start element address, (B) no more than a end element address, and (C) an integer increment of a carry number starting from the starting element address;
   2) instructing means for sending a instruction concurrently to all memory elements;
   3) matching request means for using a required state for matching concurrently at all memory elements; and
   4) finding means for concurrently finding the enabled memory elements whose match bit output has been positively asserted; and
h) instruction means for receiving and carrying out instructions at the external bus connections of the apparatus, further comprising:
   1) command means for translating the content at the external bus connections of the apparatus into exclusive means and concurrent means;
   2) result means for presenting the result of the command means at the external bus connections of the apparatus.

[Claim 26] An apparatus of [Claim 25], the required state of matching of each of its memory elements being predefined.

[Claim 27] An apparatus of [Claim 25], the instructing means of its matching means further comprising condition means for concurrently sending the required state of matching to all memory elements;

[Claim 28] An implementation of the finding means of the concurrent means of apparatus of [Claim 25], in which the state of any disabled memory element is never in any required state for matching.

[Claim 29] An implementation of the finding means of the concurrent means of apparatus of [Claim 25], in which the state of any disabled memory element can be in a required state for matching but all disabled memory element are excluded from the finding means.

[Claim 30] An apparatus of [Claim 25], further comprising:
a) a parallel counter, comprising:
   1) a plurality of bit inputs,
   2) a count output, and
   3) means for concurrently counting the bit inputs which are positively asserted;
b) the finding means of its concurrent means further comprising counting means for concurrently counting the enabled memory elements whose match bit outputs are positively asserted.

[Claim 31] Steps for using the apparatus of [Claim 30].
a) steps for concurrently finding none of the enabled memory elements which satisfies a matching requirement; and

b) steps for concurrently counting the enabled memory elements each of which satisfies a matching requirement.

[Claim 32] An apparatus of [Claim 25], further comprising:

a) a priority encoder, comprising:
   1) a plurality of bit inputs, each of which corresponds to a unique address;
   2) a no-hit bit output, which is positively asserted only when none of the bit inputs is positively asserted;
   3) a priority high bit input; and
   4) an address output, when the no-hit bit output being negatively asserted, the address output containing either (A) the highest address of the bit inputs which are positively asserted when the priority high bit input is positively asserted, or (B) the lowest address of the bit inputs which are positively asserted when the priority high bit input is negatively asserted; and
b) the finding means of its concurrent means further comprising addressing means for finding either (1) the highest or (2) the lowest element address of the enabled memory elements whose match bit outputs are positively asserted.

[Claim 33] Steps for using the apparatus of [Claim 32].

a) steps for concurrently finding none of the enabled memory elements satisfies a matching requirement;
b) steps for concurrently finding the highest address of all the enabled memory elements each of which satisfy a matching requirement;
c) steps for concurrently finding the lowest address of all the enabled memory elements each of which satisfy a matching requirement; and
d) steps for enumerating the addresses of the enabled memory elements each of which satisfies a matching requirement.

[Claim 34] An apparatus of [Claim 32], further comprising:

a) a parallel counter, comprising:
   1) a plurality of bit inputs,
   2) a count output, and
   3) means for concurrently counting the bit inputs which are positively asserted;
b) the finding means of its concurrent means further comprising counting means for concurrently counting the enabled memory elements whose match bit outputs are positively asserted.

[Claim 35] Steps for using the apparatus of [Claim 34]:

1) steps for concurrently finding none of the enabled memory elements satisfies a matching requirement;
2) steps for concurrently finding the highest address of all the enabled memory elements which satisfy a matching requirement;
3) steps for concurrently finding the lowest address of all the enabled memory elements which satisfy a matching requirement; and
4) steps for enumerating the addresses of the enabled memory elements each of which satisfies a matching requirement; and

5) steps for concurrently counting the enabled memory elements satisfies a matching requirement.

[Claim 36] An apparatus of [Claim 34], further comprising:

a) a plurality of bit storage elements;
b) means for connecting each enable bit input of all the memory elements from a unique bit storage element; and
c) a general decoder, comprising:
   1) a start address input;
   2) an end address input;
   3) a carry number input;
   4) a plurality of bit outputs, each of which has a unique address; and
   5) means for concurrently positively asserting all the bit outputs whose addresses are: (A) no less than the value at the start address input, (B) no more than the value at the end address input, and (C) an integer increment of the value at the carry number input starting from the value at the start address input, while negatively asserting all the other bit outputs;
d) a 2-stage switch, comprising a enabling stage and a dividing stage;
e) when the 2-stage switch is at the enabling stage, the enabling means for its concurrent means further comprising means for connecting:
   1) each of all bit storage elements from a unique bit output of the general decoder with the address of the bit output of the general decoder being identical to the element address of the enable bit that connects the bit storage element;
   2) its match bit output of each of all the memory elements to a unique bit input of the parallel counter, and
   3) its match bit output of each of all the memory elements to a unique bit input of the priority encoder with the element address of its match bit output being identical to the address of the bit input of the priority encoder; or
f) when the 2-stage switch is at the dividing stage, its concurrent means further comprising dividing means for obtaining (1) the quotient and (2) the value of dividend minus reminder, of dividing a dividend by a divider, the dividend being the value of a subtrahend minus an offset, the dividing means further comprising:
   1) means for inputting the offset into the start address input of the general decoder;
   2) means for inputting the subtrahend to the end address input of the general decoder;
   3) means for inputting the divider to the carry number input of the general decoder;
   4) means for connecting each of all bit outputs of the general decoder to a unique bit input of the parallel counter, except the bit output at address 0 of the general decoder;
   5) means for outputting the quotient from the count output of the parallel counter;
   6) means for connecting each of all bit outputs of the general decoder to a unique bit input the priority encoder with the address of the bit output of the general decoder being identical to the address of the bit input of the priority encoder, except (A) the bit output at address 0 of the general decoder, and (B) negatively asserting the bit input at address 0 of the priority encoder;
   7) means for positively asserting the priority high bit input of the priority encoder;

8) when the no-hit bit output of the priority encoder is positively asserted, means for signaling the divider being 0; and
9) when the no-hit bit output of the priority encoder is negatively asserted, means for outputting the value of dividend minus reminder from the address output of the priority encoder; and

g) its instruction means further comprising means for obtaining (1) the quotient, and (2) the value of dividend minus reminder, of dividing a dividend by a divider, the dividend being the value of a subtrahend minus an offset.

[Claim 37] An apparatus of [Claim 25], further comprising:

a) each of its memory elements further comprising:
   1) one addressable register;
   2) its instruction input further comprising:
      A) a datum; and
      B) a comparison code, specifying the condition of either (i) equal, or (ii) unequal, or (iii) greater than, or (iv) less than, or (v) greater than or equal, or (vi) less than or equal;
   3) a value comparator, comparing the datum of its instruction input and the content of its addressable register; and
   4) its matching means further comprising means for positively asserting its match bit output only when the result of value comparison between the datum of its instruction input and the content of its addressable register meets the comparison code of its instruction input; and
b) its concurrent means further comprising means for concurrently finding any value match between a value and the content of the addressable register of each enabled memory element.

[Claim 38] An apparatus of [Claim 41], further comprising:

a) each of its memory elements further comprising:
   1) its instruction input further comprising a mask;
   2) means for masking the content of its addressable register with the mask datum of its instruction input; and
   3) means for inputting the masked content of its addressable register into the value comparator; and
b) its concurrent means further comprising means for concurrently finding any value match between a value and the masked content of the addressable register of each enabled memory element.

[Claim 39] Steps for using apparatus of [Claim 37] to store and manage an array contains values, further comprising:

a) steps for storing each value to be compared in each memory element;
b) steps for concurrently defining an new array within an existing array;
c) steps for concurrently finding none of the array item satisfying a value matching requirement;
d) steps for concurrently finding the array item which (A) satisfies a value matching requirement, and (B) has the highest index;

e) steps for concurrently finding the array item which (A) satisfies a value matching requirement, and (B) has the lowest index;
f) steps for concurrently enumerating the array items each of which satisfies a value matching requirement;
g) steps for concurrently counting the array items each of which satisfies a matching requirement;
h steps for finding value bound to the array by concurrent steps;
i) steps for finding global extreme values of the array by concurrent steps; and
j) steps for constructing a histogram of the array by concurrent steps.

[Claim 40] An apparatus comprising:

a) a plurality of memory elements, each of which comprising:
   1) at least one register, comprising:
      A) at least one addressable register; and
      B) a neighboring register;
   2) an instruction input, comprising a select bit;
   3) executing means for carrying out instructions for the memory element from its instruction input;
   4) an enable bit input;
   5) enabling means for only enabling the executing means when the enable bit input is positively asserted;
   6) a unique element address;
   7) neighboring means for reading the neighboring register of the neighboring memory element whose element address is either (A) immediately lower when the select bit of its instruction input is negatively asserted, or (B) immediately higher when the select bit of its instruction input is positively asserted.
   8) a match bit output;
   9) state means for defining states for the memory element; and
   10) matching means for positively asserting its match bit output only when the memory element is in a required state;
b) an input/output control unit, comprising:
   1) a external bus connection, and
   2) means for communicating with the external bus;
c) exclusive means for exclusively copying either (1) the content of any addressable register to its exclusive bus, or (2) the content of its exclusive bus to any addressable register;
d) concurrent means for concurrently executing a same instruction in all its enabled memory elements, the concurrent means further comprising:
   1) enabling means for concurrently positively asserting the enable bit inputs of all memory elements whose element addresses are (A) no less than a start element address, (B) no more than a end element address, and (C) an integer increment of a carry number starting from the starting element address;
   2) instructing means for sending a instruction concurrently to all memory elements;
   3) matching request means for using a required state for matching concurrently at all memory elements; and

4) finding means for concurrently finding the enabled memory elements whose match bit output has been positively asserted; and

h) instruction means for receiving and carrying out instructions at the external bus connections of the apparatus, further comprising:
   1) command means for translating the content at the external bus connections of the apparatus into exclusive means and concurrent means;
   2) result means for presenting the result of the command means at the external bus connections of the apparatus.

[Claim 41] An apparatus of [Claim 40], further comprising:

a) each of its memory elements further comprising:
   1) one addressable register;
   2) its instruction input further comprising:
      A) a datum;
      B) a comparison code, specifying the condition of either (i) equal, or (ii) unequal, or (iii) greater than, or (iv) less than, or (v) greater than or equal, or (vi) less than or equal;
      C) a self bit; and
      D) a transfer bit;
   3) a value comparator, comparing the datum of its instruction input and the content of its addressable register; and
   4) asserting means for asserting one bit of its neighboring register when the memory element is enabled, the bit being its match bit of the memory element, the asserting means further comprising:
      A) when the self bit of its instruction input is positively asserted, positively asserting its match bit only when the result of value comparison between the datum of its instruction input and the content of its addressable register meets the comparison code of its instruction input; or
      B) when (i) the self bit of its instruction input is negatively asserted, and (ii) the transfer bit of its instruction input is negatively asserted, positively asserting its match bit only when (i) the result of value comparison between the datum of its instruction input and the content of its addressable register meets the comparison code of its instruction input, and (ii) the match bit of its neighboring memory element is positively asserted; or
      C) when (i) the self bit of its instruction input is negatively asserted, (ii) the transfer bit of its instruction input is positively asserted, and (iii) its match bit is positively asserted, copying the value of the match bit of its neighboring memory element to its match bit; or
      D) when (i) the self bit of its instruction input is negatively asserted, (ii) the transfer bit of its instruction input is positively asserted, and (iii) its match bit is negatively asserted, positively asserting its match bit only when the result of value comparison between the datum of its instruction input and the content of its addressable register meets the comparison code of its instruction input;
   5) the matching means further comprising means for positively asserting its match bit output only when the match bit is positively asserted; and

b) its concurrent means further comprising:

1) means for concurrently finding any value match between (A) a value and (B) the content of the addressable register of each enabled memory element; and
2) means for concurrently finding any value match between (A) a value and (B) the combined content of addressable registers of each enabled neighboring memory elements.

[Claim 42] An apparatus of [Claim 41], further comprising:

a) each of its memory elements further comprising:
   1) its instruction input further comprising a mask;
   2) means for masking the content of its addressable register with the mask datum of its instruction input; and
   3) means for inputting the masked content of its addressable register into the value comparator;
b) its concurrent means further comprising:
   1) means for concurrently finding any value match between (A) a value and (B) the masked content of the addressable register of each enabled memory element; and
   2) means for concurrently finding any value match between (A) a value and (B) the combined masked content of addressable registers each enabled neighboring memory elements.

[Claim 43] Steps for using apparatus of [Claim 41] to store and manage arrays, further comprising:

a) steps for storing each value to be compared either (1) in each memory element or (2) in each neighboring memory elements;
b) steps for concurrently defining an array within an existing array;
c) steps for concurrently finding none of the array item satisfying a value matching requirement;
d) steps for concurrently finding the array item which (1) satisfies a value matching requirement, and (2) has the highest index;
e) steps for concurrently finding the array item which (1) satisfies a value matching requirement, and (2) has the lowest index;
f) steps for concurrently enumerating the array items each of which satisfies a value matching requirement;
g) steps for concurrently counting the array items each of which satisfies a value matching requirement;
h steps for finding value bound to array by concurrent steps;
i) steps for finding global extreme values of array by concurrent steps; and
j) steps for constructing a histogram of the array by concurrent steps.

[Claim 44] An apparatus of [Claim 40], further comprising:

a) each of its memory element further comprising:
   1) an operation register;
   2) an status register comprising a status bit;
   3) a plural of data registers;
   4) its instruction input further comprising:
      A) a datum;

B) an operand code, encoding for either (i) the datum of its instruction input, or (ii) its neighboring register, or (iii) the neighboring register of its neighboring memory element, or (iv) any of its data registers;
C) an operation code;
D) a comparison code, encoding for either (i) equal, or (ii) unequal, or (iii) greater than, or (iv) less than, or (v) greater than or equal, or (vi) less than or equal, or (vii) any; and
E) a status code bit; and
F) a logic code;

5) operand selecting means for selecting an operand according to its operand code of its instruction input;
6) operation means for carrying out the operation specified by the operation code of its instruction input, the operation means further comprising:
A) means for positively asserting its match bit output;
B) means for positively asserting the status bit of its status register;
C) means for negatively asserting the status bit of its status register;
D) means for copying the content of its operand to its operation register;
E) means for copying the content of its operation register to either (i) its neighboring register, or (ii) any of its data registers;
7) a value comparator comparing its operand and its operation register;
8) a logic control unit, comprising:
A) means for positively asserting a comparison logic bit only when the value relation between its operand and its operation register meets the comparison code of its instruction input;
B) means for positively asserting a status logic bit only when the status bit of its status register and the status code bit of its instruction input are identically asserted; and
C) condition means for enabling the operation means only when the logic combination of (i) the comparison logic bit and (i) the status logic bit meets the logic code of its instruction input;

b) its concurrent means further comprising:
1) means for concurrently performing any conditional matching operations within each of all enabled memory element;
2) means for concurrently performing any content moving operations; and
3) means for concurrently exchanging the contents of neighboring memory elements.

[Claim 45] An apparatus of [Claim 44], further comprising:

a) each of its memory element further comprising:
1) its instruction input further comprising:
A) an operation bit section code; and
B) an operand bit section code;
2) its operand selecting means further comprising means for selecting the bit section of its operand according to the operand bit section code of its instruction input; and

3) operation selecting means for selecting the bit section of its operation register according to the operation bit section code of its instruction input for its operation means;

b) its concurrent means further comprising means for concurrently performing any conditional matching operations on sub-register level within each of all enabled memory element.

[Claim 46] An apparatus of [Claim 44], each of its memory element further comprising:

a) its status register comprising a plural of status bit;
b) its instruction input further comprising a plural of status code bit, with each status code bit corresponding to a status bit of its status register;
c) its logic control unit further comprising
   1) a plural status logic bit, with each status logic bit corresponding to a status bit of its status register;
   2) means for positively a status logic bit only when the corresponding status bit and the corresponding status code bit are identically asserted; and
   3) enabling means for enabling the operation means only when the logic combination of (i) the comparison logic bit and (i) the status logic bits meets the logic code of its instruction input.

[Claim 47] An apparatus of [Claim 44], the logic control unit of each of its memory element further comprising:

a) means for positively a AND logic bit only when all bits of its operand are positively asserted;
b) means for positively a OR logic bit only when any bit of its operation register is positively asserted; and
c) enabling means for enabling the operation means only when the logic combination of (i) the comparison logic bit, (i) the status logic bit, (iii) the AND logic bit, and (iv) the OR logic bit meets the logic code of its instruction input.

[Claim 48] Steps for using apparatus of [Claim 44] to store and manage arrays, further comprising:

a) steps for storing array either (1) with each array in each memory element or (2) each array in each neighboring memory elements or (3) with each memory elements contains a fixed number of array items;
b) steps for concurrently defining a new array in an existing array;
c) steps for concurrently finding none of the array item satisfying a matching requirement;
d) steps for concurrently finding the array item which (1) satisfies a matching requirement, and (2) has the highest index;
e) steps for concurrently finding the array item which (1) satisfies a matching requirement, and (2) has the lowest index;
f) steps for concurrently enumerating the array items each of which satisfies a matching requirement;

g) steps for concurrently counting the array items each of which satisfies a matching requirement;
h) steps for concurrently finding local extreme values of the array;
i) steps for finding value bound to array by concurrent steps;
j) steps for finding global extreme values of the array by concurrent steps;
k) steps for constructing a histogram of the array by concurrent steps;
l) steps for concurrently inserting a new array item anywhere in the array while keeping the array memory allocation closely packed;
m) steps for concurrently deleting a existing array item anywhere in the array while keeping the array memory allocation closely packed;
n) steps for concurrently exchanging two existing neighboring array items anywhere in the array while keeping the array memory allocation closely packed;
o) steps for concurrently sorting the array, and
p) steps for concurrently a new array item properly into a sorted array.

[Claim 49] An apparatus of [Claim 44], each of its memory element further comprising:
a) an increment register; and
b) its operation means further comprising (1) means for incrementing the increment register and (2) means for resetting the increment register.

[Claim 50] Steps for using apparatus of [Claim 49], further comprising:
a) steps for separating a matching requirement into steps of matching requirements;
b) steps for concurrently incrementing the increment register of each of all enabled memory elements which meets a step of the match requirement; and
c) steps for using the value of the increment register of each of all enabled memory elements as a degree indicator for meeting the matching requirement.

[Claim 51] An apparatus of [Claim 44], further comprising:
a) each of its memory element further comprising:
    1) its status register further comprising a carry bit;
    2) a parallel adder, operating on its operation register and its operand, saving the operation result to its operation register and the carry bit of the status register;
    3) its operation means further comprising :
        A) means for masking its operation register by bitwise AND with its operand;
        B) means for masking its operation register by bitwise AND with the bitwise NOT of its operand;
        C) means for masking its operation register by bitwise OR with its operand;
        D) means for masking its operation register by bitwise OR with the bitwise NOT of its operand;
        E) means for masking its operation register by bitwise XOR with its operand;
        F) means for adding its operand to its operation register;
        G) means for subtracting its operand from its operation register;
        H) means for positively asserting the carry bit of the status register; and
        H) means for negatively asserting the carry bit of the status register; and

4) the condition means of its logic control unit further comprising means for enabling its operation means only when the logic combination of (i) the comparison logic bit, (ii) the status logic bit and (iii) the carry bit of the status register meets the logic code of its instruction input;

b) its concurrent means further comprising:
   1) logic means for concurrently performing any conditional bitwise logic operations within each of all enabled memory element; and
   2) addition/subtraction means for concurrently performing any conditional addition/subtraction arithmetic operations within each of all enabled memory element.

[Claim 52] An apparatus of [Claim 51], the logic control unit of each of all memory element further comprising:

a) means for performing a logic operation on all selected bit(s) of either the operation register or the operand of the memory element to obtain a corresponding logic bit; and
b) its enabling means further comprising means for enabling the operation means only when the logic combination of (i) the comparison logic bit, (ii) the status logic bit, (iii) the carry bit of the operation bit and (iv) all logic bits meets the logic code of its instruction input.

[Claim 53] An apparatus of [Claim 51], further comprising:

a) each of its memory element further comprising:
   1) its instruction input further comprising:
      A) an operation bit section code; and
      B) an operand bit section code, encoding for the bit section of its operand
   2) its operand selecting means further comprising means for selecting the bit section of its operand according to the operand bit section code of its instruction input; and
   3) operation selecting means for selecting the bit section of its operation register according to the operation bit section code of its instruction input for the operation means;
b) its concurrent means further comprising:
   1) bitwise logic means for concurrently performing any conditional matching operations on any sub-register level within each of all enabled memory element;
   2) bitwise addition/subtraction means for concurrently performing any conditional addition/subtraction arithmetic operations on any sub-register level within each of all enabled memory element.
   3) arithmetic steps for concurrently performing any conditional arithmetic operations within each of all enabled memory element using the bitwise logic means and the bitwise addition/subtraction means.

[Claim 54] An apparatus of [Claim 51], further comprising an instruction kernel storing predefined steps of the logic means and the addition/subtraction means.

[Claim 55] Steps for using apparatus of [Claim 51] to store and manage arrays, further comprising:

a) steps for storing array either (1) with each array in each memory element or (2) each array item in each neighboring memory elements or (3) with each memory elements contains affixed number of array items;
b) steps for concurrently defining a new array within an existing;
c) steps for concurrently finding none of the array item satisfying a matching requirement;
d) steps for concurrently finding the array item which (1) satisfies a matching requirement, and (2) has the highest index;
e) steps for concurrently finding the array item which (1) satisfies a matching requirement, and (2) has the lowest index;
f) steps for concurrently enumerating the array items each of which satisfies a matching requirement;
g) steps for concurrently counting the array items each of which satisfies a matching requirement;
h) steps for concurrently finding local extreme values of the array;
i) steps for finding value bound to array by concurrent steps;
j) steps for finding global extreme values of the array by concurrent steps;
k) steps for constructing a histogram of the array by concurrent steps;
l) steps for concurrently inserting a new array item anywhere in the array while keeping the array memory allocation closely packed;
m) steps for concurrently deleting a existing array item anywhere in the array while keeping the array memory allocation closely packed;
n) steps for concurrently exchanging two existing neighboring array items anywhere in the array while keeping the array memory allocation closely packed;
o) steps for concurrently sorting the array,
p) steps for concurrently inserting a new array item properly into a sorted array;
q) steps for numerical characterization of the array by concurrent steps;
r) steps for concurrently finding the degree of matching for a matching requirement;
s) steps for concurrently carrying out a local logic and arithmetic operation involve neighboring array items;
t) steps for concurrently matching a template against neighboring array items of the array;
u) steps for concurrently carrying out certain global logic and arithmetic operations that treat each array items equally.

[Claim 56] An apparatus comprising:

a) a plurality of memory elements, each of which comprising:
    1) at least one register, comprising:
        A) at least one addressable register;
        B) a neighboring register;
    2) an instruction input, comprising:
        A) a X select bit; and
        B) a Y select bit;

3) executing means for carrying out instructions for the memory element from its instruction input;
4) an enable bit input;
5) enabling means for only enabling its executing means when its enable bit input is positively asserted;
6) a unique element address, comprising
   A) a X address; and
   B) a Y address;
7) neighboring means for reading the neighboring register of its neighboring memory element, the neighboring means further comprising:
   A) when (i) the X select bit of its instruction input is negatively asserted, and (ii) the Y select bit of its instruction input is negatively asserted, means for reading the neighboring register of its neighboring memory element (i) whose X address of the element address is immediately lower, and (ii) whose Y address of the element address is immediately lower;
   B) when (i) the X select bit of its instruction input is positively asserted, and (ii) the Y select bit of its instruction input is negatively asserted, means for reading the neighboring register of its neighboring memory element (i) whose X address of the element address is immediately higher, and (ii) whose Y address of the element address is immediately lower;
   C) when (i) the X select bit of its instruction input is negatively asserted, and (ii) the Y select bit of its instruction input is positively asserted, means for reading the neighboring register of its neighboring memory element (i) whose X address of the element address is immediately lower, and (ii) whose Y address of the element address is immediately higher; and
   D) when (i) the X select bit of its instruction input is positively asserted, and (ii) the Y select bit of its instruction input is positively asserted, means for reading the neighboring register of its neighboring memory element (i) whose X address of the element address is immediately higher, and (ii) whose Y address of the element address is immediately higher;
8) a match bit output;
9) state means for defining states for the memory element; and
10) matching means for positively asserting its match bit output only when the memory element is in a required state;

b) an input/output control unit, comprising:
   1) a external bus connection, and
   2) means for communicating with the external bus;
c) exclusive means for exclusively copying either (1) the content of any addressable register to its exclusive bus, or (2) the content of its exclusive bus to any addressable register;
d) concurrent means for concurrently executing a same instruction in all its enabled memory elements, the concurrent means further comprising:
   1) enabling means for concurrently positively asserting the enable bit inputs of all neighboring memory elements whose element addresses form a two-dimensional pattern by their X addresses and Y addresses, the two-dimensional pattern being the enabling pattern;

2) instructing means for sending a instruction concurrently to all memory elements;
3) matching request means for using a required state for matching concurrently at all the enabled memory elements; and
3) finding means for concurrently finding the enabled memory elements whose match bit output has been positively asserted; and

h) instruction means for receiving and carrying out instructions at the external bus connections of the apparatus, further comprising:
1) command means for translating the content at the external bus connections of the apparatus into exclusive means and concurrent means;
2) result means for presenting the result of the command means at the external bus connections of the apparatus.

[Claim 57] An apparatus of [Claim 56]

a) a plurality of bit storage elements;
b) means for connecting each enable bit input of all the memory elements from a unique bit storage element; and
c) the enabling means of its concurrent means further comprising means for using the bit storage elements to positively assert each corresponding enable bit input of all the memory elements; and
d) means for writing the bit storage elements with a predefined enabling pattern.

[Claim 58] An apparatus of [Claim 58], further comprising means for concurrently copying the value of match bit output of each of all enabled memory element to the corresponding bit storage elements, so that the existing enabling pattern.

[Claim 59] An apparatus of [Claim 56], further comprising:

a) a X range decoder and a Y range decoder, each range decoder comprising:
1) a start address input;
2) an end address input;
3) a plurality of bit outputs, each of which has a unique address; and
4) means for concurrently positively asserting all the bit outputs whose addresses are: (A) no less than the value at the start address input, and (B) no more than the value at the end address input, while negatively asserting all the other bit outputs; and

b) the enabling means of its concurrent means further comprising XY means for asserting the enable bit of each of all memory elements by a logic combination of:
1) a bit output from the X range decoder with the address of the bit output being identical to the X address of the element address; and
2) a bit output from the Y range decoder with the address of the bit output being identical to the Y address of the element address

[Claim 60] A preferred implementation of apparatus of [Claim 59], its XY means of the enabling means of its concurrent means further comprising means for combining the bit output from the X range decoder and the bit output from the Y range decoder by logic AND for each memory element.

[Claim 61] An apparatus of [Claim 56], further comprising:

a) each of its memory element further comprising:
   1) an operation register;
   2) an status register comprising
      A) a status bit; and
      B) a carry bit;
   3) a plural of data registers;
   4) its instruction input further comprising:
      A) a datum;
      B) an operand code, encoding for either (i) the datum of its instruction input, or (ii) the neighboring register of this memory element, or (iii) the neighboring register of its neighboring memory element, or (iv) any of the data registers;
      C) an operation code;
      D) a comparison code, encoding for either (i) equal, or (ii) unequal, or (iii) greater than, or (iv) less than, or (v) greater than or equal, or (vi) less than or equal, or (vii) any; and
      E) a status code bit; and
      F) a logic code;
   5) operand selecting means for selecting an operand according to its operand code of its instruction input;
   6) a parallel adder, operating on its operation register and its operand, and saving the operation result to its operation register and the carry bit of the status register; and
   7) operation means for carrying out the operation specified by the operation code of its instruction input, the operation means further comprising:
      A) means for positively asserting its match bit output;
      B) means for positively asserting the status bit of its status register;
      C) means for negatively asserting the status bit of its status register;
      D) means for copying the content of its operand to its operation register;
      E) means for copying the content of its operation register to either (i) its neighboring register, or (ii) any of its data registers;
      F) means for performing bitwise logic NOT of its operand;
      G) means for performing bitwise logic AND of (i) the content of operation register and (ii) its operand
      H) means for performing bitwise logic AND on (i) the content of operation register and (ii) the bitwise logic NOT on its operand
      I) means for performing bitwise logic AND on (i) the content of operation register and (ii) its operand
      J) means for performing bitwise logic AND on the content of operation register and the bitwise logic NOT on its operand
      K) means for performing bitwise logic XOR on the content of operation register and its operand
      L) means for adding its operand to its operation register;
      M) means for subtracting its operand from its operation register;
      N) means for positively asserting the carry bit of the status register; and
      O) means for negatively asserting the carry bit of the status register;
   8) a value comparator comparing its operand and its operation register;

9) a logic control unit, comprising:
   A) comparison means for positively asserting a comparison logic bit only when the value relation between its operand and its operation register meets the comparison code of its instruction input;
   B) status means for positively asserting a status logic bit only when the status bit of the status register and the status code bit of its instruction input are identically asserted; and
   C) enabling means for enabling the operation means only when the logic combination of (i) the comparison logic bit, (ii) the status logic bit and (iii) the carry bit of its status register meets the logic code of its instruction input; and

b) its concurrent means further comprising:
   1) means for concurrently performing conditional complicated matching operations;
   2) means for concurrently performing any content moving operations;
   3) means for concurrently exchanging the contents of neighboring memory elements;
   4) means for concurrently performing conditional complicated bitwise logic operations within each of all enabled memory element;
   5) means for concurrently performing conditional complicated addition/subtraction arithmetic operations within each of all enabled memory element; and
   6) means for concurrently performing conditional complicated arithmetic operations within each of all enabled memory element using the logic means and the addition/subtraction means.

[Claim 62] An apparatus of [Claim 61], the logic control unit of each of all memory element further comprising:

a) means for performing a logic operation on all selected bit(s) of either its operation register or its operand of the memory element to obtain a corresponding logic bit; and
b) its enabling means further comprising means for enabling the operation means only when the logic combination of (i) the comparison logic bit, (ii) the status logic bit, (iii) the carry bit of the operation bit and (iv) each logic bit meets the logic code of its instruction input.

[Claim 63] Steps for using apparatus of [Claim 61] to store and manage arrays, further comprising:

a) steps for storing array either (1) with each array in each memory element or (2) each array item in each neighboring memory elements or (3) with each memory elements contains a fixed number of array items;
b) steps for concurrently defining a new array of the same dimension within an existing array;
c) steps for concurrently defining a row array within an existing two dimensional array;
d) steps for concurrently defining a column array within an existing two dimensional array;
e) steps for concurrently finding none of the array item satisfying a matching requirement;
f) steps for concurrently enumerating the array items each of which satisfies a matching requirement;

g) steps for concurrently counting the array items each of which satisfies a matching requirement;
h) steps for concurrently finding local extreme values of the array;
i) steps for finding value bound to array by concurrent steps;
j) steps for finding global extreme values of the array by concurrent steps;
k) steps for constructing a histogram of the array by concurrent steps;
l) steps for statistical characterization of the array by concurrent steps;
m) steps for concurrently finding the degree of matching for a matching requirement;
n) steps for concurrently carrying out a local logic and arithmetic operation involve neighboring array items;
o) steps for concurrently matching a template against neighboring array items of the array;
p) steps for concurrently carrying out certain global logic and arithmetic operations that treat each array items equally.

[Claim 64] Steps for using apparatus of [Claim 61] in image processing, further comprising:
a) steps for storing image with (1) data for each pixel in each memory elements and (2) with the pattern of the image enclosing the pattern formed by enabled memory elements;
b) steps for concurrently defining a pattern of pixels within an existing image;
c) steps for concurrently finding none of the pattern pixel value in the pattern satisfying a matching requirement;
d) steps for concurrently enumerating the pixels in the pattern each of which satisfies a matching requirement;
e) steps for concurrently counting the pixels in the pattern each of which satisfies a matching requirement;
f) steps for constructing a histogram of the pattern pixel values by concurrent steps;
g) steps for statistical characterization of the pattern pixel values by concurrent steps;
h) steps for finding value bound to the pattern pixel values by concurrent steps;
i) steps for concurrently finding the local extreme values of the pattern pixel values;
j) steps for concurrently enlarge or shrinking the pattern through its boundary;
k) steps for concurrently ranking the pattern pixels;
l) steps for concurrently finding the degree of matching for a matching requirement in the pattern;
m) steps for concurrently carrying out local image filters;
n) steps for concurrently carrying out local template matching; and
o) steps for concurrently detecting edge limes.

[Claim 65] An apparatus comprising:
a) a plurality of memory elements, each of which comprising:
    1) at least one register, further comprising:
        A) at least one addressable register, and
        B) at least one neighboring register;
    2) an instruction input;

3) executing means for carrying out instructions for the memory element from its instruction input;
4) an enable bit input;
5) enabling means for only enabling the executing means when the enable bit input is positively asserted;
6) a match bit output;
7) state means for defining states for the memory element;
8) matching means for positively asserting its match bit output only when the memory element is in a required state;
9) a unique element address; and
10) neighboring means for reading the neighboring register of the neighboring memory element whose element address is immediately lower;

b) an input/output control unit, comprising:
1) a external bus connection, and
2) means for communicating with the external bus;

c) exclusive means for exclusively copying either (1) the content of any addressable register to its exclusive bus, or (2) the content of its exclusive bus to any addressable register;

d) concurrent means for concurrently executing a same instruction in all its enabled memory elements, the concurrent means further comprising:
1) enabling means for concurrently positively asserting the enable bit inputs of all its memory elements whose element addresses are with in an address range;
2) instructing means for sending a instruction concurrently to all memory elements;
3) matching request means for using a required state for matching concurrently at all memory elements; and
4) finding means for concurrently finding the enabled memory elements whose match bit output has been positively asserted; and

e) instruction means for receiving and carrying out instructions at the external bus connections of the apparatus, further comprising:
1) command means for translating the content at the external bus connections of the apparatus into exclusive means and concurrent means;
2) result means for presenting the result of the command means at the external bus connections of the apparatus.

[Claim 66] An equivalent implementation of apparatus of [Claim 65], the neighboring means of each of all memory elements consisting means for reading the neighboring register of the neighboring memory element whose element address is immediately higher instead lower;.

[Claim 67] An apparatus of [Claim 65], further comprising:

a) a range decoder, comprising:
1) a start address input;
2) an end address input;
3) a plurality of bit outputs, each of which has a unique address; and

4) means for concurrently positively asserting all the bit outputs whose addresses are: (A) no less than the value at the start address input, and (B) no more than the value at the end address input, while negatively asserting all the other bit outputs;

b) the enabling means of its concurrent means further comprising means for using the range decoder to implement the enabling of all memory elements within an address range between a start address and an end address.

[Claim 68] An apparatus of [Claim 65], further comprising:

a) each of its memory elements further comprising:
   1) its instruction input further comprising:
      A) a datum; and
      B) a self code bit;
   2) an equal comparator, comparing the datum of the instruction and the content of the addressable register;
   4) asserting means for asserting one bit of its neighboring register when the memory element is enabled, the bit being its match bit of the memory element, the asserting means further comprising:
      A) when the self code bit of its instruction input is positively asserted, positively asserting its matching bit only when the datum of its instruction input is identical to the content of the addressable register; or
      B) when the self bit of its instruction input is negatively asserted, positively asserting its match bit only when (A) the datum of its instruction input is identical to the content of the addressable register, and (B) the match bit of its neighboring memory element is positively asserted; and
   5) the matching means further comprising means for positively asserting its match bit output only when the match bit is positively asserted; and

b) its concurrent means further comprising:
   1) means for finding any equal match between (A) a datum and (B) the content of the addressable register of each of all enabled memory element; and
   2) means for finding any equal match between (A) a stream of datum and (B) the combined content of addressable registers of all neighboring memory elements which have been enabled;

[Claim 69] An apparatus of [Claim 68], further comprising:

a) each of its memory elements further comprising:
   1) its instruction input further comprising a mask;
   2) means for masking the content of the addressable register with the mask of its instruction input; and
   3) means for comparing the masked content of the addressable register with the datum of its instruction input in equal comparator; and

b) its concurrent means further comprising:
   1) means for finding any equal match between (A) a datum and (B) the masked content of the addressable register of each of all enabled memory element; and
   2) means for finding any equal match between (A) a stream of datum and (B) the combined masked content of addressable registers of all neighboring memory elements which have been enabled;

[Claim 70] An apparatus of [Claim 68], further comprising:

a) each of its memory elements further comprising:
   1) its instruction input further comprising an equal bit; and
   2) the writing means further comprising:
      A) when (A) the self bit of its instruction input is positively asserted, and (B) the equal bit of its instruction input is positively asserted, positively asserting its match bit only when the datum of its instruction input is identical to the content of the addressable register; or
      B) when (A) the self bit of its instruction input is negatively asserted, and (B) the equal bit of its instruction input is positively asserted, positively asserting its match bit only when (A) the datum of its instruction input is identical to the content of the addressable register, and (B) the match bit of its neighboring memory element is positively asserted; or
      C) when (A) the self bit of its instruction input is positively asserted, and (B) the equal bit of its instruction input is negatively asserted, positively asserting its match bit only when the datum of its instruction input is different from the content of the addressable register; or
      D) when (A) the self bit of its instruction input is negatively asserted, and (B) the equal bit of its instruction input is negatively asserted, positively asserting its match bit only when (A) the datum of its instruction input is different from the content of the addressable register, and (B) the match bit of its neighboring memory element is positively asserted; and

b) its concurrent means further comprising:
   1) means for finding any equal/unequal match between (A) a datum and (B) the content of the addressable register of each of all enabled memory element; and
   2) means for finding any equal/unequal match between (A) a stream of datum and (B) the combined content of addressable registers of all neighboring memory elements which have been enabled.

[Claim 71] An apparatus of [Claim 68], further comprising:

a) each of its memory elements further comprising:
   1) its instruction input further comprising a transfer bit; and
   2) the writing means further comprising:
      A) when the self bit of its instruction input is positively asserted, positively asserting its match bit only when result of equal comparison between the datum of its instruction input and the content of the addressable register meets the matching requirement; or
      B) when (A) the self bit of its instruction input is negatively asserted, and (B) the transfer bit of its instruction input is positively asserted, transferring the value of the match bit of its neighboring memory element to its match bit; and

b) its concurrent means further comprising:
   1) means for finding any equal match between (A) a datum and (B) the content of the addressable register of each of all enabled memory element;

2) means for finding any equal match between (A) a stream of datum and (B) the combined content of addressable registers of all neighboring memory elements which have been enabled; and
3) means for finding any equal match between (A) a stream of datum which contains unspecified datum and (B) the combined content of addressable registers of all neighboring memory elements which have been enabled.

[Claim 72] Searching steps for searching from data stored in the addressable registers of an apparatus of [Claim 68], for a value to be searched which has several portions, with each portion spanning an addressable, the searching steps further comprising:

a) steps for storing the data to be search in the apparatus in the proper order of the element address;
b) steps for concurrently defining or concurrently changing the selection of the enabled memory elements for searching;
c) steps for positively asserting the match bit of each of all the memory elements when the content of the addressable register of the memory element equals the first portion of the value to be matched;
d) in the order from the first portion to the last portion of the value to be searched, steps for positively asserting the match bit of each of all the memory elements when: (1) the content of the addressable register of the memory element equals the corresponding portion of the value to be matched; and (2) the match bit of its neighboring memory element of immediately previous order has positively asserted; and
e) steps for using its match bit output to signal the memory element which contains the last portion of each of all the neighboring memory elements which together hold a datum that matches the value to be searched.

Claim Structure

[Claim 1]: broadest independent claim.

[Claim 1]→[Claim 2]: parallel processing of concurrent means and exclusive means.

[Claim 1]→[Claim 2]→[Claim 3]: task switching.

[Claim 1]→[Claim 4]: termination means for f the instruction means.

[Claim 1]→[Claim 5]: a command bit input.

[Claim 1]→[Claim 5]→[Claim 6]: compliance means.

[Claim 1]→[Claim 5]→[Claim 6]→[Claim 7]: preferred compliance means.

[Claim 1]→[Claim 5]→[Claim 6]→[Claim 8]: use of compliance means.

[Claim 1]→[Claim 9]: with instruction micro-kernel.

[Claim 1]→[Claim 10]: preferred maximal carry number.

[Claim 1]→[Claim 11]: implementation by general decoder.

[Claim 1]→[Claim 12]: carry number to be constant 1;

[Claim 1]→[Claim 10]→[Claim 13]: implementation by range decoder.

[Claim 14]: [Claim 1] plus neighboring connections.

[Claim 14]→[Claim 15]: simplest content movable memory.

[Claim 14]→[Claim 15]→[Claim 16]: content movable memory that stores data from lowest address.

[Claim 14]→[Claim 15]→[Claim 17]: content movable memory that stores data from highest address.

[Claim 14]→[Claim 15]→[Claim 18]: preferred implementation of content movable memory.

[Claim 14]→[Claim 15]→[Claim 19]: dynamic content movable memory with refresh capability.

[Claim 14]→[Claim 15]→[Claim 19]→[Claim 20]: dynamic content movable memory with automatic refresh capability.

[Claim 14]→[Claim 15]→[Claim 19]→[Claim 21]: use dynamic content movable memory with refresh capability to replace static random accessible memory.

[Claim 14]→[Claim 15]→[Claim 22]: content movable memory with management capability.

[Claim 14]→[Claim 15]→[Claim 22]→[Claim 23]: content movable memory with containment.

[Claim 14]→[Claim 15]→[Claim 22]→[Claim 24]: content movable memory with error detection and correction capability.

[Claim 25]: [Claim 1] plus matching.

[Claim 25]→[Claim 26]: predefined matching requirement.

[Claim 25]→[Claim 27]: specified matching requirement.

[Claim 25]→[Claim 28]: implementing matching in favor of construct cost.

[Claim 25]→[Claim 29]: implementing matching in favor of energy cost.

[Claim 25]→[Claim 30]: matching by parallel counter.

[Claim 25]→[Claim 30]→[Claim 31]: use of matching by parallel counter.

[Claim 25]→[Claim 32]: matching by priority encoder.

[Claim 25]→[Claim 32]→[Claim 33]: use of matching by priority encoder.

[Claim 25]→[Claim 32]→[Claim 34]: matching by priority encoder and parallel counter.

[Claim 25]→[Claim 32]→[Claim 34]→[Claim 35]: use of matching by priority encoder and parallel counter.

[Claim 25]→[Claim 32]→[Claim 34]→[Claim 36]: additional functionality of parallel divider.

[Claim 25]→[Claim 37]: simplest content comparable memory.

[Claim 25]→[Claim 38]: maskable content comparable memory.

[Claim 25]→[Claim 39]: use of simplest content comparable memory.

[Claim 40]: [Claim 1] plus (a) neighboring connections and (b) matching.

[Claim 40]→[Claim 41]: content comparable memory with neighboring connections.

[Claim 40]→[Claim 41]→[Claim 42]: maskable content comparable memory with neighboring connections.

[Claim 40]→[Claim 41]→[Claim 43]: use of content comparable memory with neighboring connections.

[Claim 40]→[Claim 44]: simplest database memory.

[Claim 40]→[Claim 44]→[Claim 45]: database memory with bit section.

[Claim 40]→[Claim 44]→[Claim 46]: database memory with multiple status bits.

[Claim 40]→[Claim 44]→[Claim 47]: database memory with AND/OR logic bits.

[Claim 40]→[Claim 44]→[Claim 48]: use of database memory.

[Claim 40]→[Claim 44]→[Claim 49]: database memory with incrementing capability.

[Claim 40]→[Claim 44]→[Claim 49]→[Claim 50]: use of database memory with incrementing capability.

[Claim 40]→[Claim 44]→[Claim 51]: math memory.

[Claim 40]→[Claim 44]→[Claim 51]→[Claim 52]: math memory with bit section.

[Claim 40]→[Claim 44]→[Claim 51]→[Claim 53]: math memory with additional logic.

[Claim 40]→[Claim 44]→[Claim 51]→[Claim 54]: math memory with instruction kernel.

[Claim 40]→[Claim 44]→[Claim 51]→[Claim 55]: use of math memory.

[Claim 56]: [Claim 1] plus (a) 2D neighboring connections and (b) matching.

[Claim 56]→[Claim 57]: predefined 2D enabling pattern;

[Claim 56]→[Claim 57]→[Claim 58]: refinement of enabling pattern by matching;

[Claim 56]→[Claim 59]: enabling pattern by range decoders.

[Claim 56]→[Claim 59]→[Claim 60]: preferred enabling pattern by range decoders.

[Claim 56]→[Claim 61]: image memory.

[Claim 56]→[Claim 61]→[Claim 62]: use of image memory.

[Claim 65]: [Claim 1] plus (a) one-side neighboring connection and (b) matching.

[Claim 65]→[Claim 66]: alternative one-side neighboring connection.

[Claim 65]→[Claim 67]: implementation by range decoder

[Claim 65]→[Claim 68]: simplest content searchable memory.

[Claim 65]→[Claim 68]→[Claim 69]: maskable content searchable memory.

[Claim 65]→[Claim 68]→[Claim 70]: content searchable memory with unequal search.

[Claim 65]→[Claim 68]→[Claim 71]: content searchable memory with search containing wild datum in a datum stream.

[Claim 65]→[Claim 68]→[Claim 72]: use of content searchable memory.